This handout includes space for every question that requires a written response. Please feel free to use it to handwrite your solutions (legibly, please). If you choose to typeset your solutions, the `README.md` for this assignment includes instructions to regenerate this handout with your typeset LaTeX solutions.

3.a

Training Metrics:

- **training/Factorization Loss:**
  - shared=False: 0.2469 (value), 0.2472 (smoothed)
  - shared=True: 0.264 (value), 0.2643 (smoothed)
  - Parameter sharing shows higher factorization losses compared to separate models.
- **training/Joint Loss:**
  - shared=False: 0.2527 (value), 0.253 (smoothed)
  - shared=True: 0.2711 (value), 0.2714 (smoothed)
  - Parameter sharing exhibits higher joint losses.
- **training/MSE:**
  - shared=False: 0.8325 (value), 0.8325 (smoothed)
  - shared=True: 0.9758 (value), 0.9746 (smoothed)
  - MSE is significantly higher with parameter sharing.

Test Metrics:

- **eval/MSE:**
  - shared=False: 0.9073 (value), 0.9073 (smoothed)
  - shared=True: 1.0179 (value), 1.0179 (smoothed)
  - Separate models show lower MSE on the test set.
- **eval/Mean Reciprocal Rank:**
  - shared=False: 0.0644 (value), 0.0646 (smoothed)
  - shared=True: 0.0468 (value), 0.0471 (smoothed)
  - Separate models demonstrate better performance on the mean reciprocal rank.

Conclusion:

Separate models outperform parameter sharing in all metrics on both training and test sets. Therefore, parameter sharing does not outperform separate models for $\lambda_F = 0.99$ and $\lambda_R = 0.01$.

3.b

Training Metrics:

- **training/Factorization Loss:**
  - shared=False: 0.247 (value), 0.2473 (smoothed)
  - shared=True: 0.4287 (value), 0.4285 (smoothed)
  - Separate models have significantly lower factorization losses.
- **training/Joint Loss:**
  - shared=False: 0.5397 (value), 0.5399 (smoothed)
  - shared=True: 0.6309 (value), 0.6308 (smoothed)
  - Separate models show lower joint losses.
- **training/MSE:**
  - shared=False: 0.8325 (value), 0.8324 (smoothed)
  - shared=True: 0.8332 (value), 0.8331 (smoothed)
  - Separate models have slightly lower MSE.

Test Metrics:

- **eval/MSE:**
  - shared=False: 0.9077 (value), 0.9076 (smoothed)
  - shared=True: 0.9066 (value), 0.9066 (smoothed)
  - Parameter sharing shows a slightly lower MSE on the test set.
- **eval/Mean Reciprocal Rank:**
  - shared=False: 0.064 (value), 0.0643 (smoothed)
  - shared=True: 0.0201 (value), 0.0202 (smoothed)
  - Separate models have significantly better mean reciprocal rank.

Conclusion:

Separate models outperform parameter sharing on most metrics on the training set. However, on the test set, the eval/MSE metric is slightly better for parameter sharing, but the eval/Mean Reciprocal Rank is significantly better for separate models. Thus, parameter sharing does not outperform separate models in general for $\lambda_F = 0.5$ and $\lambda_R = 0.5$.

3.c

Results for $\lambda_F = 0.99$ and $\lambda_R = 0.01$:

- **training/Factorization Loss:**
  - Value: 0.264
  - Smoothed: 0.2643
- **training/Joint Loss:**
  - Value: 0.2711
  - Smoothed: 0.2714
- **training/MSE:**
  - Value: 0.9758
  - Smoothed: 0.9746
- **eval/MSE:**
  - Value: 1.0179
  - Smoothed: 1.0179
- **eval/Mean Reciprocal Rank:**
  - Value: 0.0468
  - Smoothed: 0.0471

Results for $\lambda_F = 0.5$ and $\lambda_R = 0.5$:

- **training/Factorization Loss:**
  - Value: 0.4287
  - Smoothed: 0.4285
- **training/Joint Loss:**
  - Value: 0.6309
  - Smoothed: 0.6308
- **training/MSE:**
  - Value: 0.8332
  - Smoothed: 0.8331
- **eval/MSE:**
  - Value: 0.9066
  - Smoothed: 0.9066
- **eval/Mean Reciprocal Rank:**
  - Value: 0.0201
  - Smoothed: 0.0202

Comparison:

- **training/Factorization Loss and Joint Loss:**
  - For $\lambda_F = 0.99$ and $\lambda_R = 0.01$, training losses are significantly lower than for $\lambda_F = 0.5$ and $\lambda_R = 0.5$. This may indicate that a higher learning factor (λ_F) and lower learning rate (λ_R) contribute to better training on these losses.
- **training/MSE:**
  - For $\lambda_F = 0.99$ and $\lambda_R = 0.01$, training MSE is higher than for $\lambda_F = 0.5$ and $\lambda_R = 0.5$. This could mean that the model fits the training data better with $\lambda_F = 0.5$ and $\lambda_R = 0.5$.
- **eval/MSE:**
  - On the test set, MSE for $\lambda_F = 0.5$ and $\lambda_R = 0.5$ is lower than for $\lambda_F = 0.99$ and $\lambda_R = 0.01$, indicating better generalization with the balanced configuration.
- **eval/Mean Reciprocal Rank:**
  - Mean reciprocal rank is better for $\lambda_F = 0.99$ and $\lambda_R = 0.01$, which may indicate more precise ranking with this configuration.

Conclusion:

$\lambda_F = 0.99$ and $\lambda_R = 0.01$ contribute to more stable and slower training, which helps the model fine-tune weights but can lead to worse generalization on test data. $\lambda_F = 0.5$ and $\lambda_R = 0.5$ lead to better generalization on test data but may not provide the same level of precision in ranking and lower training losses.